AFRL-IF-RS-TR-2005-127
Final Technical Report
April 2005

DETECTING AND SURVIVING LARGE-SCALE NETWORK INFRASTRUCTURE ATTACKS

University of Michigan

Sponsored by
Defense Advanced Research Projects Agency
DARPA Order No. H558 & J032

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

The views and conclusions contained in this document are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of the Defense Advanced Research Projects Agency or the U.S. Government.

AIR FORCE RESEARCH LABORATORY
INFORMATION DIRECTORATE
ROME RESEARCH SITE
ROME, NEW YORK

STINFO FINAL REPORT

This report has been reviewed by the Air Force Research Laboratory, Information Directorate, Public Affairs Office (IFOIPA) and is releasable to the National Technical Information Service (NTIS). At NTIS it will be releasable to the general public, including foreign nations.

AFRL-IF-RS-TR-2005-127 has been reviewed and is approved for publication

APPROVED: /s/

ALAN J. AKINS
Project Engineer

FOR THE DIRECTOR: /s/

WARREN H. DEBANY, JR., Technical Advisor
Information Grid Divison
Information Directorate

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 074-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing this collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503

1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	APRIL 2005	Final Jun 99 – Jul 04

4. TITLE AND SUBTITLE
DETECTING AND SURVIVING LARGE-SCALE NETWORK INFRASTRUCTURE ATTACKS

5. FUNDING NUMBERS
C - F30602-99-1-0527
PE - 62301E
PR - H558
TA - 10
WU - 01

6. AUTHOR(S)
Farnam Jahanian

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
University of Michigan
Division of Research Development and Administration
3003 South State Street
Ann Arbor Michigan 48109-1274

8. PERFORMING ORGANIZATION REPORT NUMBER
N/A

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Defense Advanced Research Projects Agency AFRL/IFGA
3701 North Fairfax Drive 525 Brooks Road
Arlington Virginia 22203-1714 Rome New York 13441-4505

10. SPONSORING / MONITORING AGENCY REPORT NUMBER
AFRL-IF-RS-TR-2005-127

11. SUPPLEMENTARY NOTES

AFRL Project Engineer: Alan J. Akins/IFGA/(315) 330-1869/ Alan.Akins@rl.af.mil

12a. DISTRIBUTION / AVAILABILITY STATEMENT
APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 Words)*

This report summarizes our work on detecting and surviving large-scale network infrastructure attacks. This work investigated several different areas of infrastructure attacks including routing protocol analysis and denial of service attacks. This work produced many significant results including several patent applications as well as the commercialization of the denial of service survival tool by Arbor Networks in 2001. In addition, we have published a number of technical research reports and presented our results in public research community venues as well as in private briefings to several government organizations.

This report presents the results of our analysis in the following areas:

- The development and use of an innovative protocol monitoring tool,
- The development and demonstration of an in-line protocol scrubber designed to remove attacks from malicious network streams,
- The design, development, and commercialization of a tool designed to detect and mitigate denial of service attacks,
- Analysis of routing protocol stability and reliability including real-world analysis of both BGP and OSPF running on production networks,
- Design and testing of an overlay network system designed to minimize network unavailability.

14. SUBJECT TERMS
Fault Tolerant Networks, Denial of Service Attacks, Cyber Defense, Intrusion Detection, Routing Protocol Analysis, Protocol Scrubber

15. NUMBER OF PAGES
36

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	UL

Table of Contents

List of Figures

1 Project Synopsis

As the financial, military, and national utility infrastructures become intertwined with that of the emerging global data networks, their stability and integrity have become synonymous. As such, detecting and surviving attacks on global network infrastructure has significant financial, military and national security ramifications. Routers and switches are among the most fundamental components of a globally scaled network. All other networking services cease to function when their connectivity is severed by a routing failure. As such, routers are significant targets for malicious parties. Unfortunately, routers and switches are themselves open to a wide range of both direct and indirect attacks. Direct attacks on the routing infrastructure include: attacks on inter-router protocols—such as Border Gateway Protocol (BGP), Open Shortest Path First (OSPF), and Routing Information Protocol (RIP)—both from a design and implementation standpoint; Denial of Service attacks on routing resources; and attacks through exploitation of inter-dependencies between a routing protocol and its underlying transport layer. Indirect attacks include sending deliberate misinformation through valid inter-router protocol messages. These latter types of attacks are the most insidious, since they would appear to the router as routine route fluctuation. Moreover, upon partitioning the network, an attacker would be free to masquerade as the severed network by usurping its nameservice and routing personae. Coordinated attacks from multiple points in a large-scale network exacerbate the problem of routing attack detection and response. However, defending the routing infrastructure from these attacks is paramount to ensuring the integrity of the network.

This effort, titled Detecting and Surviving Large-Scale Network Infrastructure Attacks (also known as the Lighthouse project) developed a distributed attack and response system for global infrastructure survivability and assurance. We focused on key challenges and critical technological advances necessary for the construction of this system. Specifically, this project has provided:

- Analysis of the stability of routing protocols under infrastructure security attacks: We performed an experimental study of operational routing protocols to uncover security weaknesses that may be exploited for large- scale infrastructure attacks. This analysis will investigate the junction between intra- and inter-domain routing protocols as well as the malicious use of interprovider routing policies to identify scenarios for subversion.
- Development of Windmill extensible passive probes: A key component of any survivable system is a scalable probe architecture for the collection of network events. The proposed probe architecture, called Windmill, utilizes passive techniques for eavesdropping on target systems. Groups of these point probes can be distributed throughout an infrastructure to provide an aggregate profile of the global network.
- Development of intelligent security gateways (ISG's) for protecting sensitive infrastructure components: Intelligent security gateways fuse the functionality of a firewall with an intrusion detection and response system. As an interposed agent, the ISG's will be used to filter sensitive network flows specific to infrastructure management and control such as inter-domain routing protocols in real-time. This real-time filtering is implemented as a two-tiered protocol scrubber. The ISG acts as a defensive countermeasure in our architecture.
- Distributed countermeasures for infrastructure survivability: Localized countermeasure approaches are ineffective in dealing with infrastructure attacks, particularly against coordinated attempts to compromise a set of routers strategically located across a global network. This research proposes a distributed countermeasure capability for isolating and neutralizing attacks on the global network infrastructure based on four increasingly sophisticated levels of response: restriction, deflection, isolation and counterattack.

- Development of overlay networks for increased network availability. Based on detailed analysis of real-world routing behavior we developed intelligent overlay network algorithms designed to intelligently place overlay nodes based on the underlying Internet Protocol (IP) topology. These placement strategies improve the availability of the overlay network while minimizing deployment and maintenance costs.
- Analysis of asymmetric flows on the Internet. This analysis identified the amount and types of asymmetry of Internet paths at both the Autonomous System (AS) and router level. In addition, we define the scope and impact of asymmetric traffic flows on Internet performance and security.

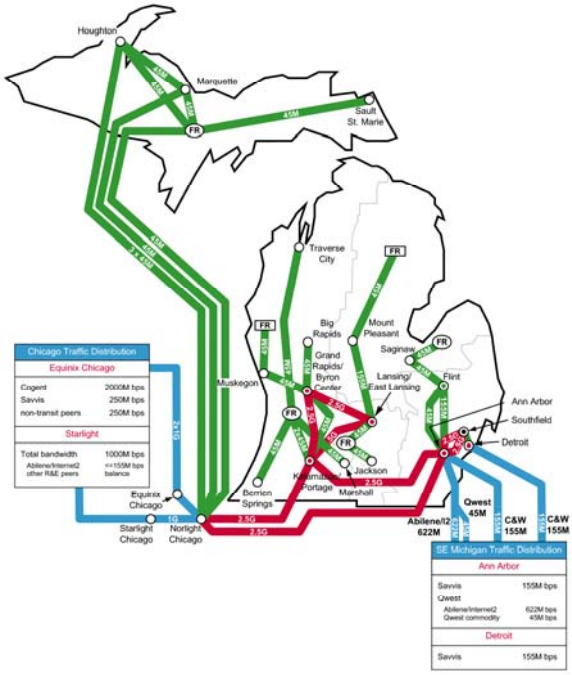


Figure 1: MichNet Topology

2 Measurement Infrastructure

In order to understand the behavior of OSPF on a production network, we connected several probe machines to MichNet routers. MichNet is a statewide network composed of hundreds of routers covering the state of Michigan (Figure 1). These routers connect over 500 physical locations to the Internet through several outside connections. 50 of these routers form the core OSPF network. The remainders are used for customer connections. In addition to external connections to Cable and Wireless and Qwest, MichNet has a high speed connection to Internet2.

3 Windmill

As a real-world system with economic incentives for uptime and robustness increasing, it is difficult to take portions of the Internet down for measurement and testing. Moreover, the rate of growth for the Internet has placed a severe tax on the network infrastructure, leaving many resources such as routers and highly trafficked Web servers in a state of constant overload. Compounding the problem is that most of the software executing the protocols is *shrink-wrapped* and is not amenable to scrutiny or modification for performance measurement—a backbone router collapses within seconds with full debugging turned on. It is precisely at these points where the performance effects of protocol interaction are the greatest, and most

poorly understood. This paper presents an architecture for an extensible software probe that can measure precisely these types of interactions under real-world conditions.

The software probe described in this work utilizes passive techniques for eavesdropping on target protocols. Groups of these point probes can be distributed throughout a target network to measure an aggregate performance profile of target protocols. Care has been taken during the probe's design to enable its placement in high bandwidth monitoring points. This allows the measurement of Internet protocols across a spectrum of vantage points, from routing exchange points and enterprise gateways to local area networks. There are many research groups involved in the deployment of Internet probe machines for the measurement of Internet paths and topologies, including the NIMI [20], Surveyor [2], and IPMA [13] projects. These projects have approached Internet measurement by utilizing active performance metrics [14] – measurements that perturb the network – such as one-way loss along a datagram stream, or periodic traceroutes between the probe and specific end-hosts. Our work complements these efforts. We have designed and implemented the architecture of a passive performance probe that can be used in conjunction with active probes – possibly housed on the same host – to measure and infer performance data from the underlying network flows without perturbing the network or infrastructure. Our probe architecture allows the measurement of high-level protocols, such as BGP, Domain Name System (DNS), and HyperText Transfer Protocol (HTTP), which sit above the Internet's base protocols. Our tool measures these protocols without modifying either the infrastructure—the routers, nameservers, Web caches, end hosts, etc.—or the host implementation of these protocols.

The architecture of our tool, Windmill, supports the passive performance measurement of application-level protocols through the use of protocol reconstruction and abstraction-breaching protocol monitoring. A probe experiment infers the end-host's view of a target protocol by recursively executing the lower protocol layers against the stream of incoming packets. Effectively, the probe reconstructs the view of the end hosts by passively monitoring the protocol's network frames. The experiments utilize interfaces exported by the probe to "lift the hood" on the lower-layer protocols, violating their abstractions to examine events and data structures that are normally hidden from the higher layers. Together, these features allow an experiment to correlate lower-layer protocol events—including checksum and length errors, packet reorderings or retransmissions, and round trip estimates—with the behavior and performance of the reconstructed application-level protocol.

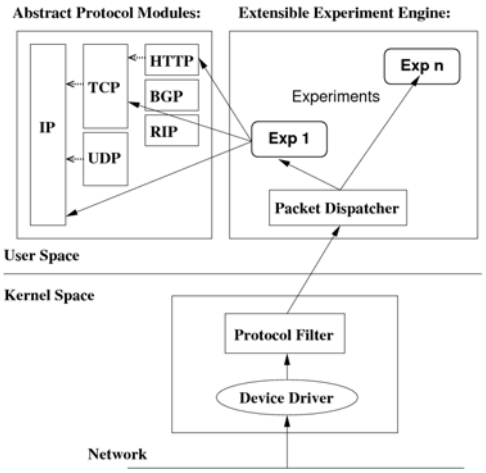


Figure 2: Organization of Windmill's architecture.

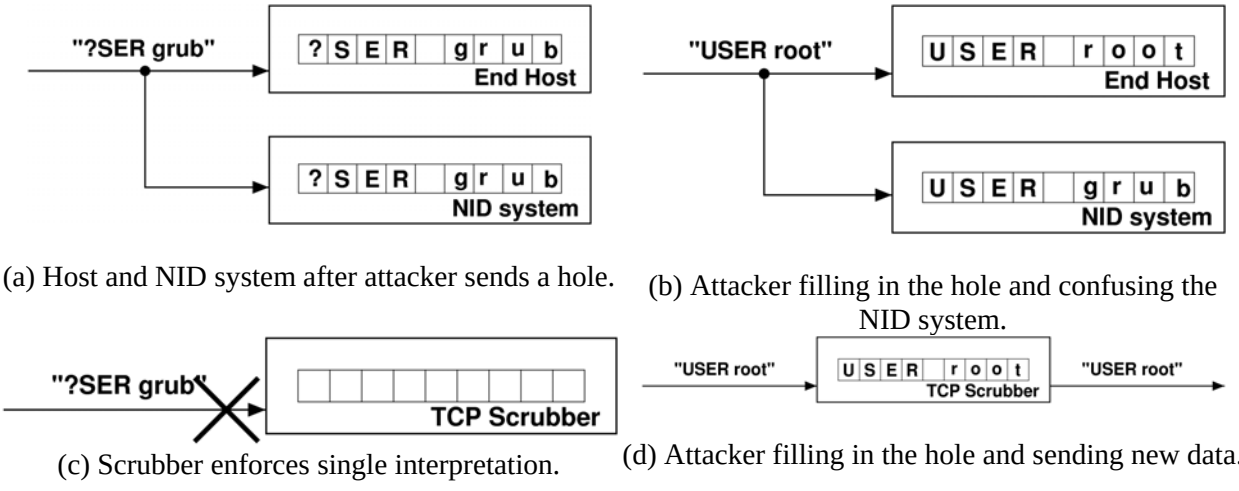
To accommodate both performance and extensibility, Windmill's software was split into three functional components: a dynamically compiled protocol filter, a set of abstract protocol modules, and an extensible experiment engine (Figure 2). Packet throughput is maximized through the use of a custom protocol filter which dynamically compiles and downloads native code into the kernel for fast multi-destination packet matching. For performance, the bulk of the user-level code is contained in a set of abstract protocol modules. These modules are C programming language implementations of the base Internet protocols. Those protocol layers that do not change – such as IP, User Datagram Protocol (UDP), Transmission Control Protocol (TCP), BGP, and HTTP – are implemented as abstract protocol modules. By calling these modules an experiment can efficiently execute a target protocol's stack on incoming matching packets. The probe's extensibility comes from the use of a custom dynamic loader that is used to load and manage the probe's experiments.

The main contributions of the Windmill probe architecture are:

- *The implementation of our passive probe architecture.* Our current implementation is built using an off-the-shelf hardware and software base. This implementation utilizes both recursive protocol reconstruction as well as abstraction violation to measure application-level Internet protocols, such as BGP, HTTP, and DNS.
- *Attention to the intrinsic tradeoff between performance and extensibility by splitting the code into two pieces.* The performance critical code—for protocol reconstruction and memory management—was placed in the tool's libraries, whereas the extensibility support was constrained to a custom runtime library. Together these pieces enable dynamic experiments to be loaded, managed, and modified over long periods of probe uptime, while allowing for the high performance protocol processing necessary for high bandwidth vantage points.
- *Creation of the fast Windmill Protocol Filter (WPF).* Since the probe is designed to execute multiple experiments simultaneously, there is the possibility that several experiments may subscribe to overlapping packet flows. In order to make matching of multiple experiments as fast as possible, this functionality was pushed into a custom packet filter. This filter utilizes dynamic compilation in conjunction with a fast matching algorithm to enable *one-to-many* packet *multiplexing* in a running time linear in the number of comparable fields for common cases. This is the same time complexity as the best most-specific *one-to-one* matching algorithms. Moreover, WPF addresses some fundamental limitations in past packet filtering technology—filters that demultiplex packets to endpoints by *most-specific* matches—by correctly handling ambiguous (overlapping) filters that do not have any natural or explicit ordering.
- *Investigation of an Internet routing instability conjecture.* The probe was used in an experiment designed to monitor and measure the BGP routing traffic exchanged between two peer border routers in order to validate one of the key observations presented in [18]. Specifically, the experiment provides a possible answer for the correlation between Internet routing instability and network bandwidth usage. The experiment suggests that the BGP protocol be modified for use with a UDP keepalive protocol.
- *A study of an Internet Collaboratory.* Windmill was used to measure the Upper Atmospheric Research Collaboratory. The experiment demonstrated the use of the tool on a real system that could not be modified for direct measurement, the use of the tool for online data reduction, and the power of using our passive tool to drive an active measurement apparatus.
- *Investigation of the stability of a campus network.* Windmill was used to perform real time analysis of the RIP routing protocol. The experiment demonstrated the use of a tool to do continuous, real time monitoring of a live network.

4 Scrubber

As society grows increasingly dependent on the Internet for commerce, banking, and mission critical applications, the ability to detect and neutralize network attacks is becoming increasingly significant. Attackers can use ambiguities in network protocol specifications to deceive network security systems. Passive entities can only notify administrators or active mechanisms after attacks are detected. However, the response to this notification may not be timely enough to withstand some types of attacks – such as attacks on infrastructure control protocols. Active modification of flows is the only way to immediately detect or prevent these attacks. This work presents the design and implementation of *protocol scrubbers*—transparent, interposed mechanisms for actively removing network attacks at various protocol layers. We describe two instances of protocol scrubbers. The *transport scrubber* addresses the problem of insertion and evasion attacks by removing protocol related ambiguities from network flows, enabling downstream passive network-based intrusion detection systems to operate with high assurance [21]. The *fingerprint scrubber* prevents a remote user from identifying the operating system of another host at the TCP/IP layers by actively homogenizing flows [34].



(a) Host and NID system after attacker sends a hole.

(b) Attacker filling in the hole and confusing the NID system.

(c) Scrubber enforces single interpretation.

(d) Attacker filling in the hole and sending new data.

Figure 3: Example of ambiguity of transport layer protocol implementation differences between an interposed agent (NID system) and an end host.

The transport scrubber's role is to convert *ambiguous* network flows—flows that may be interpreted differently at different endpoints—into *well-behaved* flows that are interpreted identically by all downstream endpoints. As an example, this report presents the implementation of a TCP scrubber that eliminates *insertion* and *evasion* attacks against passive network-based intrusion detection systems. Insertion and evasion attacks use ambiguities in protocol specifications to subvert detection. This paper argues that passive network intrusion detection systems (NID systems) can only effectively identify malicious flows when used in conjunction with an interposed active mechanism. Through interposition, the transport scrubber can guarantee consistency, enabling downstream intrusion detection systems to work with confidence. The specifications for Internet protocols allow *well-behaved* implementations to exchange packets with deterministic results. However, sophisticated attackers can leverage subtle differences in protocol implementations to wedge attacks past the NID system's detection mechanism by purposefully creating ambiguous flows. In these attacks, the destination endpoint reconstructs a malicious interpretation, whereas the passive NID system's protocol stack interprets the flow of packets as a benign

exchange. Examples of these sources of ambiguity are IP fragment reconstruction and the reassembly of overlapping out-of-order TCP byte sequences. The role of the transport scrubber is to pick one interpretation of the protocols and to convert incoming flows into a single representation that all endpoints will interpret identically (Figure 3). The transport scrubber's conversion of ambiguous network flows into consistent flows is analogous to that of network traffic shaping. Shapers modify traffic around the edges of a network to generate predictable utilization patterns within the network. Similarly, the transport scrubber intercepts packet flows at the edges of an interior network and modifies them in such a way that their security attributes are predictable.

Ambiguities in protocol specifications also allow attackers to determine a remote host's operating system. The process of determining the identity of a host's operating system by analyzing packets from that host is called TCP/IP stack fingerprinting. Fingerprinting scans are often preludes to further attacks, and therefore we built the fingerprint scrubber to block the majority of stack fingerprinting techniques in a general, fast, and transparent manner. Freely available tools (such as `nmap` [7]) exist to scan TCP/IP stacks efficiently by quickly matching query results against a database of known operating systems. The reason this is called "fingerprinting" is therefore obvious; this process is similar to identifying an unknown person by taking his or her unique fingerprints and finding a match in a database of known fingerprints. A malicious use of fingerprinting techniques is to construct a database of IP addresses and corresponding operating systems for an entire network. When someone discovers a new exploit, the attacker can now target only those machines running the vulnerable operating system. This facilitates the systematic installation of malicious code, such as distributed Denial of Service tools, on many machines without detection. Current fingerprinting techniques provide fine-grained determination of an operating system. For example, `nmap` has knowledge of many individual versions of Linux. Almost every system connected to the Internet is vulnerable to fingerprinting, including standard computers running the major operating systems, routers, switches, hubs, bridges, embedded systems, printers, firewalls, web cameras, and even some game consoles. Many of these systems, such as routers, are important parts of the Internet infrastructure, and compromised infrastructure is a more serious problem than compromised end hosts. Therefore a general mechanism to protect any system is needed.

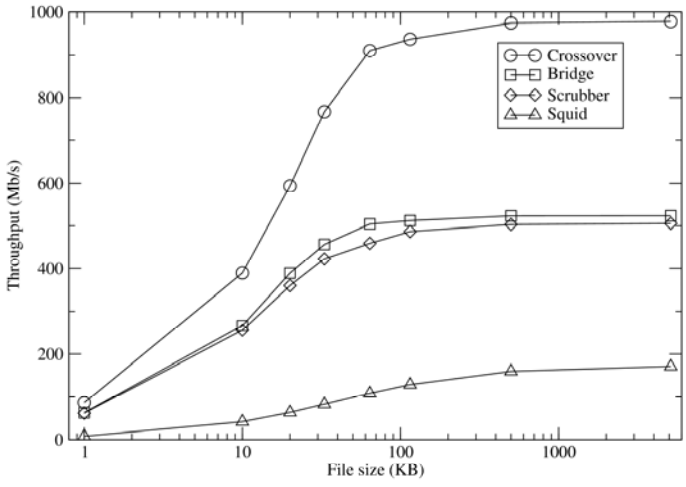


Figure 4: Gigabit TCP scrubber performance results.

The main contributions of this work are:

- *Introduction of transport scrubbing:* The paper introduces the use of an active, interposed transport scrubber for the conversion of ambiguous network flows into well-behaved, unequivocally interpreted flows. We argue that the use of a transport scrubber is essential for correct operation of passive NID systems. This section described the use of transport scrubbers to eliminate insertion and evasion attacks on NID systems [25]. The concept of transport scrubbing can easily be merged with existing firewall technologies to provide the significant security benefits outlined in this report.
- *Design and implementation of TCP scrubber:* The novel design and efficient implementation of the *half-duplex* TCP scrubber was presented. The current implementation of the TCP scrubber exists as a modified FreeBSD kernel [6]. This implementation is shown to scale with raw Unix-based Ethernet bridging (Figure 4). By keeping the design of the scrubber general, we plan to migrate the implementation to programmable networking hardware such as the Intel Internet Exchange Architecture (IXA) [26, 15].
- *Design and implementation of fingerprint scrubber:* Building upon the TCP scrubber, we presented a tool to defeat TCP/IP stack fingerprinting. The fingerprint scrubber is transparently interposed between the Internet and the network under protection. We showed that the tool blocks the majority of known stack fingerprinting techniques in a general, fast, and transparent manner.

5 DOS Attack Recognition and Backtracking Software

Coordinated Denial of Service (DoS) attacks are considered one of the most significant threats jeopardizing the explosive growth of the Web. The growing reliance of businesses and consumers on the Web for accessing information and for conducting transactions has made continuous availability of enterprise networks and web sites extremely critical.

The Internet provides a means for connecting a global set of distributed users with network services. Many of these services are *open services*—services that are accessible by anyone. A company's e-commerce Web server is a common example of an open service. Other examples include a company's incoming email, netnews, domain name service, File Transfer Protocol (FTP) servers, etc. These open services are accessible by anyone with a network address—one of the cornerstones of the Internet model. However, as open services they are equally open to abuse. Coordinated Denial of Service attacks overwhelm these open services with illegitimate traffic, denying access to real users.

The widely publicized Denial of Service attacks in February 2000 brought down numerous major Web sites including Yahoo, ZDNet, eBay, Amazon.com, CNN.com, and E-Trade. These coordinated attacks flooded the Web sites with mock traffic from computers around the globe, essentially blocking access to legitimate requests from customers for several hours. Less publicized attacks happen everyday around the clock, and there is no end in sight. A recent report predicts that security breaches in the form of corporate espionage, cyberactivism and cyberterrorism will become more prevalent. Approximately 2000 Web sites offer free attack tools, and everyday three new software exploits are discovered. It is expected that Denial of Service attacks will become increasingly common.

Outbound Denial of Service attacks are also a significant problem for network service providers and their customers. Several industry reports have recently highlighted the growing concerns about potential liability and legal implications for companies (as well as their Internet service providers) whose computers and networks are used to launch Denial of Service attacks on corporate targets. According to Giga Information Group, legal arguments based on negligence or breach of contract will form the foundation for these cases.

Denial of Service attacks generally involve organizations whose computers and networks have been compromised. The attacker first scans millions of computers on the Internet to identify unsecured hosts to

be used as “launch pads.” He then secretly installs software on a master computer and a collection of compromised “zombie” computers. The attacker hides his true identity and location by using these zombie machines to launch the attack. He only needs to have the master signal the zombies to begin the attack, specifying the attack’s target and type. The result of this malicious activity is the Denial of Service to legitimate users because the coordinated attacks often:

- Overwhelm the target Web server with illegitimate requests, thereby choking off the sites available bandwidth,
- Clog the victim’s network infrastructure (routers, proxy servers, etc.), or
- Crash the victim’s Web server.

The Internet’s open-service model is significantly threatened by Denial of Service attacks. Clearly, a means for detecting, tracking, and blocking Denial of Service attacks is critical to ensuring around-the-clock availability of open Internet services.

5.1 Our Solution

To combat the problems with Denial of Service attacks, we developed a system that detects, traces and recommends filters to counter DoS attacks destined for or originating from:

- a customer’s link,
- an ISP’s web hosting service,
- or a customer’s collocated service.

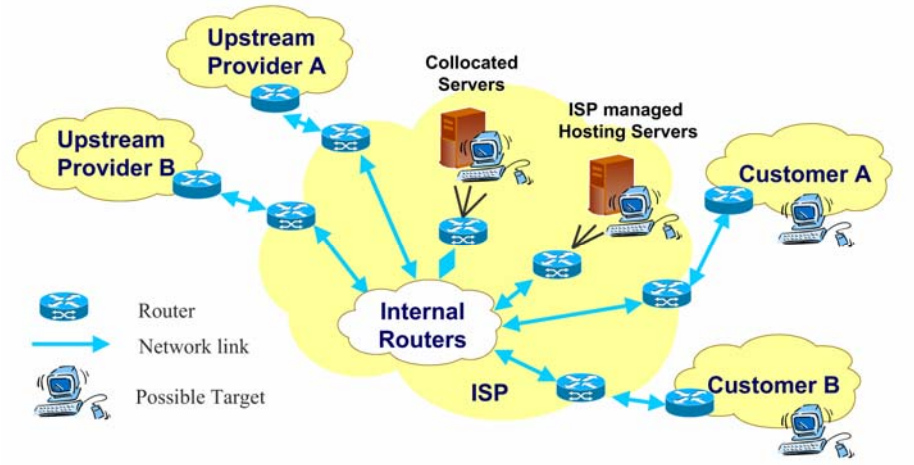


Figure 5: Potential targets of DoS attacks

With this system in place, a network engineer can be notified of any bandwidth anomalies in their network. This provides the network engineer with the ability to react quickly and efficiently to these anomalies, enabling sustained service availability. The tool pinpoints the path of the anomaly through the ISP’s network, including relevant routers and affected interfaces. At each point in the anomaly’s path, statistics are included to describe the scope and severity of the attack. With this knowledge in hand, the network engineer can quickly change the network’s operating parameters to restore connectivity and avoid customer complaints.

5.2 Features

Our unique software approach to security combines network topology information and coarse-grained traffic statistics from routers to detect, backtrack, and filter distributed attacks originating from and destined for downstream customer networks and Web sites. This first-ever solution exploits information from routers without requiring significant changes to the existing Internet routing infrastructure. Our approach leverages existing functionality that is already incorporated into the major routing vendors—such as Cisco and Juniper Networks—products.

Our solution works with existing network infrastructure, to detect, trace and filter DoS and next generation attacks:

- **Detect:** A new patent-pending process for real-time monitoring, detection, and notification of Denial of Service attacks and network anomalies. Continuous or periodic sampling is employed for collecting network statistics and extracting network topology information from routers.
- **Trace:** A new protocol for correlating anomalous distributed events that enables tracking a Denial of Service attack back to its source. This patent-pending technology allows for the application of our technology to a very large-scale network.
- **Filter:** Provides immediate attack-specific filter recommendations and profile / fingerprints. We can stop attacks at the network border to minimize their impact, as well as enable filtering out of outbound attacks to reduce liability.

5.3 Detect

Our tool offers a revolutionary mechanism for identifying Denial of Service attacks within an Internet Service Provider (ISP), a Web Hosting service, or an enterprise network. It combines a network's dynamic profile with internal static signatures of Denial of Service attacks to instantly identify malicious traffic. This patent-pending technology utilizes custom algorithms to identify Denial of Service attacks in the reams of incoming traffic flow statistics gathered from the routing infrastructure.

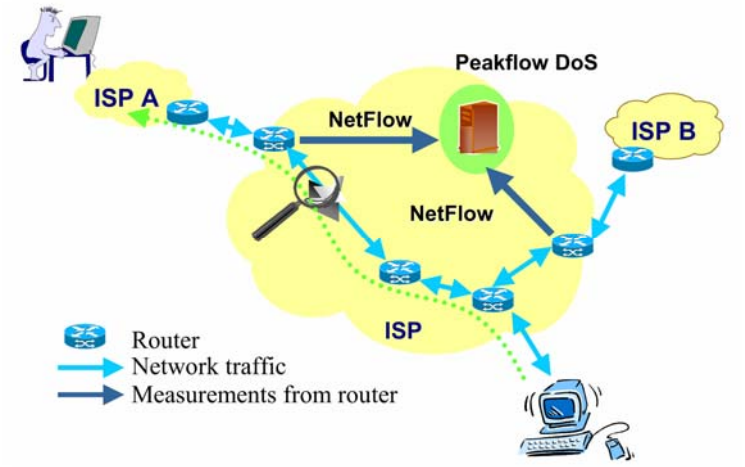


Figure 6: Tracing a forged packet. In this example, an ISP's hosting service is under attack from attackers transiting through upstream provider ISP A.

Figure 6 demonstrates how our system detects an attack. A host in ISP A is bombarding a target server in the web hosting service with a Denial of Service attack. However, the attacker is forging the return address on the packets in the attack, making it impossible to determine their true origin. The analysis engine receives flow statistics from the routers in the target's hosting service. From these statistics, it can detect the attack at each of the affected routers along its path. This path leads directly from the target to ISP A's border, where the attack originates. This example demonstrates the utility of our tool deployed within a web hosting service's network. It can also be used in both source and transit networks.

When employed at an attacker's originating network, our tool can pinpoint the location of the attacker. In this case, it will back trace the attack directly to its source's first-hop router. It may be that the attacker is a zombie residing on a compromised machine in an ISP's Web server cluster or a customer's collocation equipment. In addition to uncovering those traditional launch pads, our tool will be instrumental in identifying attacks originating from home machines that connect to the Internet through persistent Abstract Syntax Description Language (ADSL) or cable modem connections.

5.4 Trace

Our software utilizes patent-pending algorithms that provide the functionality for tracking anonymous Denial of Service attacks to their sources. These algorithms provide two main functions: directed searching and path reconstruction. *Directed searching* is an algorithm for quickly separating the attack traffic from the legitimate network traffic. By narrowing the scope of the upstream detection points, directed search provides the means for scalable tracking of large-scale attacks. *Path reconstruction* takes multiple measurements of distributed Denial of Service attacks and determines their global topology characteristics. Specifically, given a huge distributed Denial of Service attack, our software allows many statistics collected from around a service provider's network to be quickly and robustly correlated to reconstruct the attack's paths.

A custom communication protocol binds these distributed detection points together. This protocol allows multiple autonomous systems to cooperate and exchange attack information, enabling a globally scoped solution. For example, if two neighboring networks wanted to share information for tracking the attack farther up or downstream, the protocol provides a standard mechanism for this information exchange.

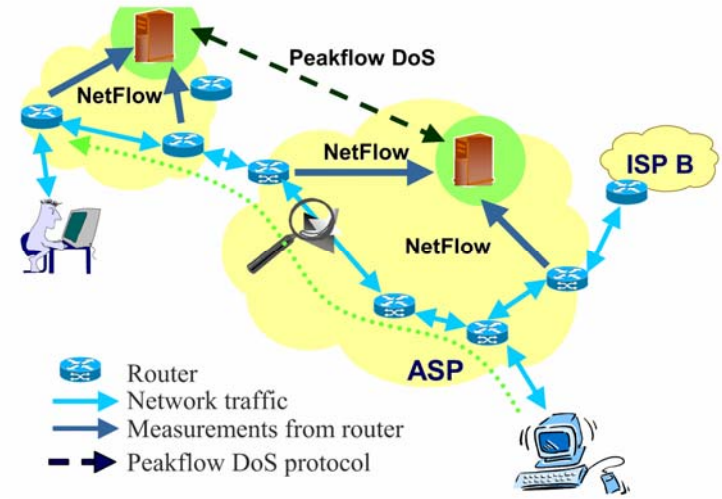


Figure 7: Tracing an attack to its source.

Figure 7 shows an example of how two systems can cooperate using our custom protocol to trace the attack to its origin.

5.5 Filter

Our system also uses attack fingerprints to suggest access control list (ACL) entries and/or committed access rate (CAR) parameters, which a network engineer can implement to filter out the attack.

As a result, our system not only alerts administrators to a developing attack, but also arms them with a detailed description, and a suggested fix. The network engineer can then change the network's operating parameters, restoring connectivity and avoiding customer complaints.

5.6 Incorporation into a Network

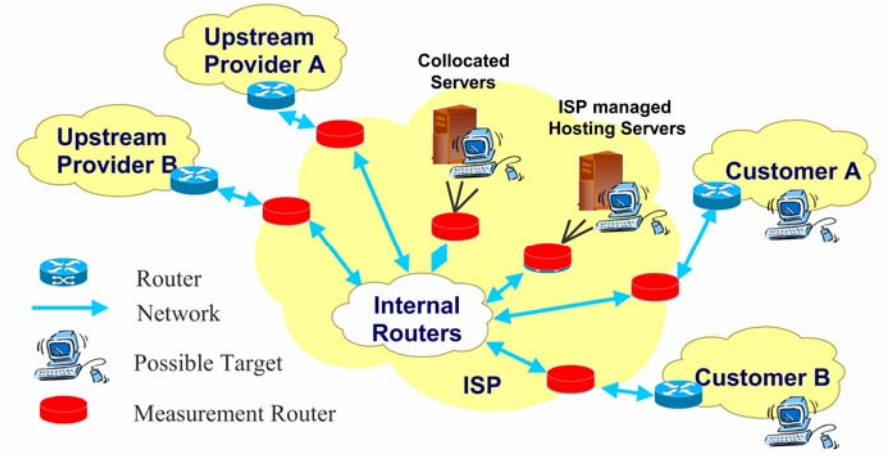


Figure 8: Example network augmented with measurement routers.

Our DoS tool can easily be incorporated into a service provider's network. The main requirement for deployment is the ability to gather flow statistics from key routers in the network—it is not necessary to collect statistics from every router. Figure 8 shows a network with the key measurement routers marked. These routers include peering points with both upstream providers and downstream customers. Additionally, this figure designates several internal routers as measurement points that provide connectivity to the ISP's managed Web servers and customer's collocation hosts. In general, a service provider would want to collect information from the routers along the network's perimeter. Our tool is non-intrusive and works with common routers and switches such as Cisco and Juniper equipment.

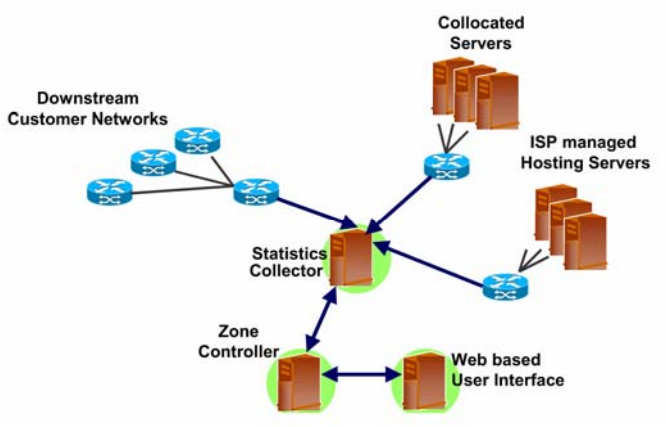


Figure 9: System architecture.

Figure 9 shows the main components in the system architecture: the statistics collector, the zone controller, and the Web based user interface. The *statistics collector* is responsible for the collection of router statistics from the measurement routers in the network. Typically, several measurement routers are collocated with a single collector. The collector is a rack-mountable system with several network interfaces and a modem.

Upon receiving statistics, the collector distills the data and transfers it to the zone controller. This transfer can take place either in-band over the network's links, or out-of-band over a Plain Old Telephone Service (POTS) line.

The *zone controller* is responsible for several statistics collectors and their associated measurement routers. The controller aggregates the data and correlates distributed events to construct a network-wide view of any bandwidth anomalies, including Denial of Service attacks. The controller consists of a high-end processing server that can be located wherever the network engineers deem convenient. After the controller identifies attack traffic, it archives the statistics in a long-term database and updates the user interface with the attack's latest statistics.

The Web based interface is used to examine both ongoing and past attacks in greater detail. The Web interface shows recent and current attacks. Past attacks can also be referenced that have been archived in the database. The Web interface shows the attacks main parameters, including its type, duration, start and end time, the attack's target, and list of affected measurement routers. Each of the routers in a specific attack can be examined further to determine the inbound interfaces affected by the attack and the specific characteristics the attack has at each of these interfaces. These include general attack parameters, such as the number of flows, packets, and bytes per second as well as specific parameters. These specific parameters are generated through a data-mining process that pinpoints the attack profile. This profile can be used to generate ACL's to filter out the attack at the interface. Finally, the Web interface allows an operator to examine a representative sample of NetFlow records that are kept for each interface. This feature allows the network operator to examine the attack's NetFlow records directly from the Web without having to telnet into the router and digging around in the NetFlow cache.

The Web interface can also be used to investigate measurement routers' configuration information. In addition the Web interface enumerates a measurement router's interface information, including network address, Simple Network Management Protocol (SNMP) index number and description string.

5.7 Summary

Our DoS system can rapidly spot attacks, closing the costly gap between the detection of a network-based security threat and its resolution. This novel, distributed anomaly-based detection scheme provides a pragmatic, highly- scalable early warning system for threats to mission-critical networks. It achieves this by regularly sampling network traffic statistics to establish a dynamic baseline of typical traffic patterns in different zones on the network. By comparing real-time network activity against this dynamic baseline, it is able to flag anomalies, including zero-day threats. Using precise fingerprints drawn from the detection and trace back processes, this technology recommends attack- and equipment-specific filters to eliminate the threat, ensuring normal operations.

The early success in demonstration and evaluation of the research concepts led to the rapid commercial availability of this technology by Arbor Networks, a network security company launched in 2001 with significant funding support from the private sector. This technology is now commercially available as Arbor's Peakflow product. Arbor's Peakflow product proactively eliminates threats against the network itself, protecting service providers as well as mission-critical government and enterprise networks.

6 Internet Routing Protocol Analysis

In a brief number of years, the Internet has evolved from an experimental research and academic network to a commodity, mission-critical component of the public telecommunication infrastructure. During this period, we have witnessed an explosive growth in the size and topological complexity of the Internet and an increasing strain on its underlying infrastructure. As the national and economic infrastructure has become increasingly dependent on the global Internet, the end-to-end availability and reliability of data networks promises to have significant ramifications for an ever-expanding range of applications. For example, transient disruptions in backbone networks that previously impacted a handful of scientists may now cause enormous financial loss and disrupt hundreds of thousands of end users.

Since its commercial inception in 1995, the Internet has lagged behind the public switched telephone network (PSTN) in availability, reliability and Quality of Service (QoS). Factors contributing to these differences between the commercial Internet infrastructure and the PSTN have been discussed in various literatures [22, 16]. Although recent advances in the Internet Engineering Task Force's (IETF's) Differentiated Services working group promise to improve the performance of application-level services within some networks, across the wide-area Internet these QoS algorithms are usually predicated on the existence of a stable underlying forwarding infrastructure.

As part of the Lighthouse project, we have performed several studies examining the stability and resiliency of Internet routing protocols. This includes real-world analysis of both Internet scale stability as well as studies of individual service provider networks. In addition, we have used experiments on live networks to measure the reaction of production networks to failure. These studies have revealed several valuable insights into the behavior of these critical networks.

6.1 Experimental Study of Internet Stability and Wide-Area Backbone Failures

Several wide spread Internet failures have led the popular press to predict the imminent "death of the Internet" [21]. Although the predicted Internet collapse has yet to materialize, further analysis of the behavior and characteristics of wide-area network faults is critical for the continued evolution of the Internet.

In this work, we describe an experimental study of Internet stability and the origins of failure in Internet protocol backbones [17]. Unlike telephony networks, the stability of end-to-end Internet paths is

dependent both on the underlying telecommunication switching system, as well as the higher level software and hardware components specific to the Internet's packet-switched forwarding, name resolution and routing architecture. Although a number of vendors provide mean-time to failure statistics for specific hardware components used in the construction of wide-area networks (e.g. power supplies, switches, etc.), estimations of the failure rates for IP backbones at a systemic level remain problematic. As we describe below, the interactions between the underlying components of the Internet are poorly understood [28].

Typical analysis of faults in telephony networks has focused on the number of customers affected by an outage [1]. The US Federal Communication Commission requires service providers to report all outages lasting 30 minutes or more and affecting 30,000 customers or more [17]. No such reporting requirements yet exist for Internet providers. And, if such requirements did exist, the same estimations of the impact of failures would be problematic for Internet providers. Both the definition of failure and even "end-user" are somewhat ambiguous on the Internet. In contrast to the fixed bandwidth used by telephony, Internet applications and end-users have widely disparate bandwidth, latency and loss requirements. For example, the failure of an Internet T3 link (45 MB) may impact one large weather simulation at a supercomputer center, or several thousand web-surfing dial-up users. In our analysis, we make no effort to quantify the significance of Internet outages based on the number of users affected. Instead, we focus on the number of individual link or interface failures, and the number of unreachable network destinations.

In general, the Internet exhibits a number of engineering and operational challenges distinct from those associated with telephony networks and applications. Most significantly, unlike switched telephony networks, the Internet is a conglomeration of thousands of heterogeneous dynamically packet switched IP backbones. No resources are explicitly reserved for each datagram or IP data flow. Instead, the end-to-end quality of Internet performance depends on the impact of loss, queuing delay and network congestion on each of the flow's individual datagram packets. So, for example, although the initial "call setup" of an Internet telephony application may succeed, all subsequent voice datagrams in the connection may be lost due to network congestion. The relationship between loss, latency and end-to-end performance remains an area of active research.

In addition, the explosive growth in demand for Internet facilities and features has resulted in a significantly more rapid Internet software and hardware evolutionary testing and development cycle than traditional amongst PSTN equipment suppliers. For example, telephony switches typically undergo development cycles on the order of several years or even decades. In contrast, some Internet backbone routers and switches have development cycles lasting six months or less. Internet vendors regularly market backbone equipment featuring new software algorithms even before these protocols have advanced into official standards [13, 24]. The technological demands associated with the Internet's growth are so severe that Internet providers often depend on these newly released products or software features to sustain their network's continued expansion. The abbreviated development cycle has led to a trade-off between reliability and time-to-market. As a result, the reliability of the Internet infrastructure has arguably suffered.

The rapid growth of IP backbones has also led to a decline in the relative level of experience and degree of coordination amongst Internet backbone operators. A number of significant recent Internet outages have stemmed from human error. Other outages have originated, or been exacerbated by lack of coordination between the backbone engineering staff of different Internet providers. In the PSTN network, a comparatively small number of telecommunication companies interact via well-defined, standardized channels using uniform management, measurement and operational procedures. The significantly more diverse and less uniform Internet does not enjoy the same degree of coordination. Specifically, the Internet lacks central administration and coordination. Unlike traditional PSTN standards bodies whose formal membership requirements are defined by international treaty [29], the only requirement for participation in the three yearly Internet standards meetings is showing up [13].

We briefly describe some recent Internet outages which directly, or indirectly, impacted a majority of Internet backbone paths. Although several major incidents stemmed from underlying PSTN failures, we focus below on faults specific to the Internet. We provide the following summaries as anecdotal evidence of the sources of major Internet failures.

- April 25, 1997: A misconfigured router maintained by a small Virginia service provider injected an incorrect routing map into the global Internet. This map indicated that the Virginia company's network provided optimal connectivity to all Internet destinations. Internet providers that accepted this map automatically diverted all of their traffic to the Virginia provider. The resulting network congestion, instability, and overload of Internet router table memory effectively shut down most of the major Internet backbones for up to two hours. Incorrect published contact information for operations staff, and lack of procedures for interprovider coordination exacerbated the problem [2].
- August 14, 1998: A misconfigured critical Internet database server incorrectly referred all queries for Internet machine names ending in ".net" to the wrong secondary database server. As a result, a majority of connections to ".net" Internet web servers and other end stations failed for a period of several hours [26].
- November 8, 1998: A malformed routing control message stemming from a software fault triggered an interoperability problem between core Internet backbone routers manufactured by different vendors. This problem lead to a persistent, pathological oscillation and failure in the communication between most Internet core backbone routers. As a result, Internet end-users experienced wide-spread loss of network connectivity, and increased packet loss and latency. The majority of backbone providers resolved the outage within several hours after adding filters which removed the malformed control message [25].

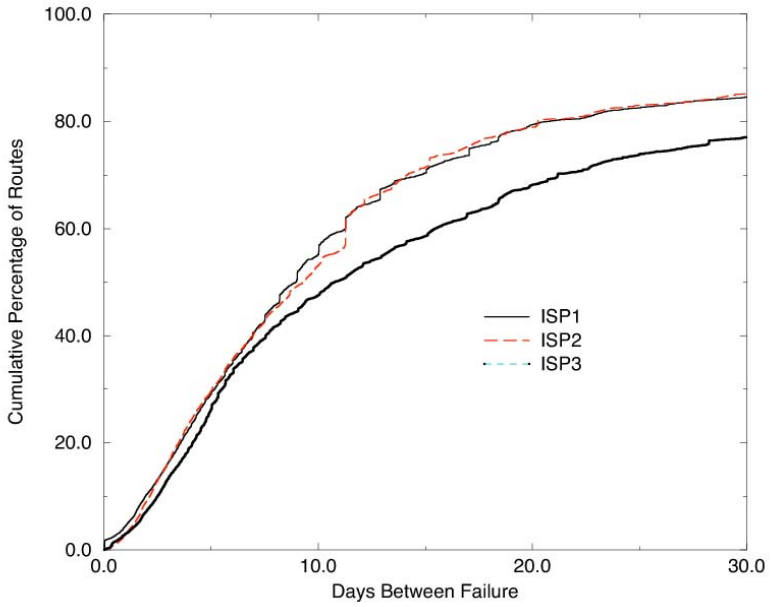


Figure 10: Cumulative distribution of the mean-time to failure for default-free routes from three ISPs.

Overall, both Internet and telephony outages stem from a wide range of sources, including faults in the underlying telecommunication switching system, and the higher level software and hardware components.

Like Pradhan [30], we are interested in estimating the reliability of Internet backbone paths at specified probability and duration thresholds such as the mean number of events per year, and the mean time spent in events. The significant findings of our work include:

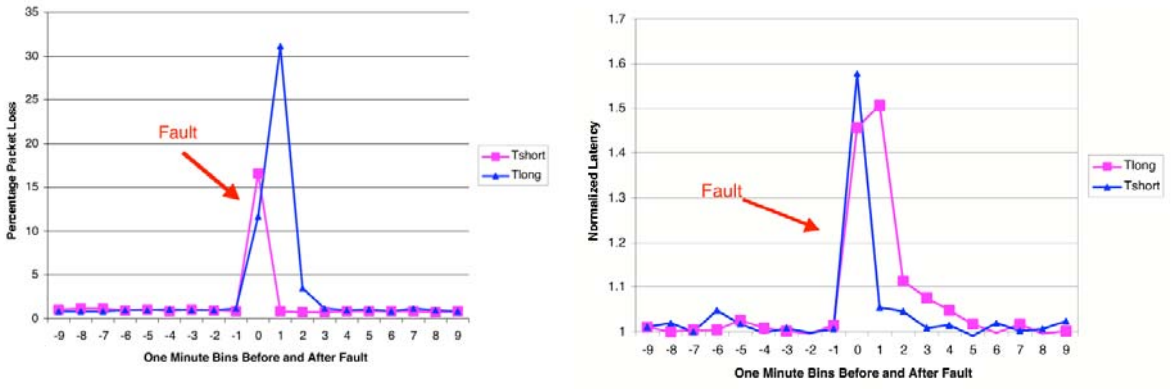
- The Internet backbone infrastructure exhibits significantly less availability and a lower meantime to failure than the Public Switched Telephone Network (PSTN).
- The majority of Internet backbone paths exhibit a mean-time to failure of 25 days or less (Figure 10), and a mean-time to repair of twenty minutes or less. Internet backbones are rerouted (either due to failure or policy changes) on the average of once every three days or less.
- Routing instability inside of an autonomous network does not exhibit the same daily and weekly cyclic trends as previously reported for routing between Inter provider backbones, suggesting that most inter-provider path failures stem from congestion collapse.
- A small fraction of network paths in the Internet contribute disproportionately to the number of long-term outages and backbone unavailability.

6.2 Delayed Internet Routing Convergence

The Internet backbone infrastructure is widely believed to support rapid restoration and rerouting in the event of individual link or router failures. At least one report places the latency of inter-domain Internet path failover on the order of 30 seconds or less based on qualitative end user experience [13]. These brief delays in inter-domain failover are further believed to stem mainly from queuing and router central processing unit (CPU) processing latencies [3, (message digests 11/98, 1/99)]. In this work, we show that most of this conventional wisdom about Internet failover is incorrect [16]. Specifically, we demonstrate that the Internet does not support effective inter-domain failover and that most of the delay in path restoral stems solely from the unexpected interaction of configurable routing protocol timers and specific router vendor protocol implementation decisions during the process of delayed BGP convergence.

The slow convergence of distance vector (DV) routing algorithms is not a new problem [20]. DV routing requires that each node maintain the distance from itself to each possible destination and the vector, or neighbor, to use to reach that destination. Whenever this connectivity information changes, the router transmits its new distance vector to each of its neighbors, allowing each to recalculate its routing table.

DV routing can take a long time to converge after a topological change because routers do not have sufficient information to determine if their choice of next hop will cause routing loops to form. The count-to-infinity problem [20] is the canonical example used to illustrate the slow convergence in DV routing. Numerous solutions have been proposed to address this issue. For example, including the entire path to the destination, known as the path vector approach, is used in the Border Gateway Protocol (BGP), the inter-domain routing protocol in the Internet. Other attempts to solve the count-to-infinity problem or accelerate convergence in many common cases include techniques such as split horizon (with poison reverse), triggered updates, and the diffusing update algorithm [7].



(a) Loss (b) Latency

Figure 11: Average percentage end-to-end loss and normalized latency of 512 byte ICMP echoes sent to 100 web sites every second during the ten minutes immediately proceeding and following the injection of a Tshort and Tlong events at the MAe-West exchange point.

In this work, we present a complimentary study of both the impact and the rate at which inter-domain repair and failure information propagates through the Internet. We also measure the impact of Internet path changes on end-to-end network performance. Specifically, our major results include:

- Although the adoption of the path vector by BGP eliminates the DV count-to-infinity problem, the path vector exponentially exacerbates the number of possible routing table oscillations.
- The delay in Internet inter-domain path failovers averaged three minutes during the two years of our study, and some percentage of failovers triggered routing table oscillations lasting up to fifteen minutes.
- The theoretical upper bound on the number of computational states explored during BGP convergence is $O(n!)$, where n is the number of autonomous systems in the Internet. We note that this is a theoretical upper bound on BGP convergence and is unlikely to occur in practice.
- If we assume bounded delay on BGP message propagation, then the lower bound on BGP convergence is $\Omega((n-3)\times 30)$ seconds, where n is the number of autonomous systems in the Internet.
- The delay of inter-domain route convergence is due almost entirely to the unforeseen interaction of protocol timers with specific router vendor implementation decisions.
- Internet path failover has significant deleterious impact on end-to-end performance—measured packet loss grows by a factor of 30 and latency by a factor of four during path restoral (Figure 11).
- Minor changes to current vendor BGP implementations would, if deployed, reduce the lower bound on inter-domain convergence time complexity from $\Omega((n-3)\times 30)$ to $\Omega(30)$ seconds, where n is the number of autonomous systems in the Internet.

6.3 Experiences With Monitoring OSPF on a Regional Service Provider Network

This study collected routing protocol data from MichNet, a mid-sized regional Internet service provider covering the state of Michigan [37]. Data was collected from four geographically distributed probe machines running custom monitoring software. This year-long effort collected information

continuously from MichNet's intra-domain routing protocol, OSPF. We used this data to analyze the performance of OSPF on a production network and discovered surprising anomalies. There is a long history of studies looking for these kinds of problems in routing protocols. Detailed studies of routing on the Internet have uncovered significant problems with the performance of the Border Gateway Protocol (BGP) [8, 18, 16, 17, 19]. BGP, an inter-domain routing protocol, interconnects the autonomous systems that comprise the Internet. As such, any significant problems with BGP can affect routing on the Internet as a whole. Unlike BGP, the work on intra-domain routing protocols such as OSPF and Intermediate System to Intermediate System (IS-IS) has mainly focused on their theoretical behavior. Distinguished by the fact that they are run within an autonomous system, intra-domain routing protocols have a significant impact on the performance of the local network. In addition, because inter-domain and intra-domain routing protocols are often interdependent, problems with one can affect the performance of the other. While these studies based on simulations and small scale testbeds have identified important issues, they still fall short of identifying the issues that affect these protocols on production networks. Our work, however, has focused on understanding the performance of intra-domain protocols under real-world conditions. We attempt to highlight several anomalies of OSPF that are not seen under controlled conditions.

Our work also complements the few existing studies that have looked at the behavior of intra-domain routing protocols on production networks. Our work extends these studies by providing year-long analysis, corroborating others' results and introducing previously unseen results. A study of Qwest's network focused on issues affecting the convergence time of IS-IS [1]. This study identified single, flapping links as the predominant source of instability. It also made the theoretical case for using an incremental shortest path algorithm. Our work corroborates this by identifying the lopsided distribution of problem links and providing practical measurements supporting the argument for an incremental shortest path algorithm. Our work also complements two studies concurrent to ours. The first paper analyzes link failures on Sprint's IS-IS backbone [12]. It examines the temporal locality and duration of link failures in addition to studying the effects of induced link failures. The second paper provides a more comprehensive study of OSPF on an enterprise network [32]. It also provides temporal and frequency views of instability in the network, and makes a distinction between internal OSPF links and external links. Finally, the paper looks at issues with redundant Link State Advertisements (LSAs) caused by OSPF's reliable forwarding mechanism. Our work presents several complimentary results to these two papers. First, we present an analysis of the amount, duration, and frequency of updates seen on MichNet. We also analyze these updates both by source and by the type of change they represent. By collecting a year's worth of routing updates we are also able to demonstrate that these results are not singular anomalies but rather common issues across time and different networks.

In addition to providing complementary results to these existing and concurrent studies, our work introduces new issues with intra-domain routing. Our analysis found significant periods of localized instability. The predominant source of this strange routing behavior was from customer networks. These routes are injected into OSPF from other routing protocols such as RIP. Like most ISPs, MichNet is constructed from around 50 core, backbone routers connecting a much larger number of customer networks. Often overlooked in routing analysis, this additional layer of hierarchy is a significant source of instability. Unlike the simplified view that would consider MichNet an autonomous system, far more of the routers within MichNet are not actually under the operators' control. These customer networks often have less monitoring, lower levels of redundancy, and often use older routing protocols to maintain their connection to MichNet. These factors all contribute to the increased level of instability we see with these routes. Some of the anomalies we see from these injected routes appear to be caused by well known failure behavior of the originating routing protocol.

We also expand the existing body of knowledge by providing detailed analysis of the source and behavior of several specific anomalies. These anomalies exhibit very surprising behavior that produces notable effects throughout our analysis. In addition to highlighting previously unobserved behavior these anomalies demonstrate the significant impact that information injected into OSPF from external routing protocols can have on the network. These specific anomalies account for the most prominent period of instability, the first and second largest source of instability by router, and the largest source of instability by an individual link. In addition, these anomalies demonstrate the need for new network management tools that understand routing protocols. All of these anomalies appear to go unnoticed by the network operators for significant periods of time. Better monitoring and management of the underlying routing protocols would improve the reliability and performance of the network.

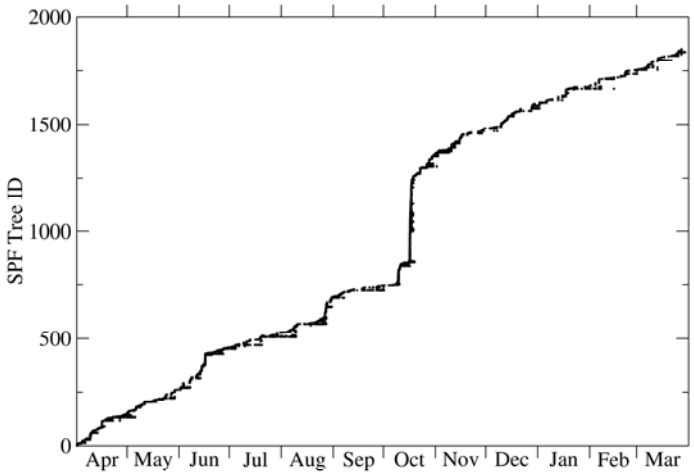


Figure 12: Configuration changes in the shortest path tree from a single router.

The main contributions of this work are:

- A detailed, year-long analysis of OSPF running on a production network. We examine the overall traffic levels as well as the amount, source, and duration of instability on the network. In addition, we examine the changes in routing topology over time. We also make an important distinction between the core OSPF network, and the edge network comprising the customer connections, leading to new insights about the behavior of OSPF. We corroborate the results seen in previous work, and discuss previously unseen issues.
- The identification of customer networks as a major source of instability injected into OSPF. While it is well known that customer networks are less stable, we show that this external instability causes increased instability in the core OSPF network.
- The identification of individually flapping links as the predominant source of instability. These high frequency oscillations are confined to a single physical link and do not induce failures in other parts of the network. In addition, several routers contribute disproportionately to this instability. We also found that 80% of the individual periods of flapping last less than five minutes.
- The discovery that the routing topology is constantly evolving although the individual changes are incremental (Figure 12). Rather than reverting back to a stable configuration, we found that no

single topology of the network lasted longer than 28 days. In addition, we show that a significant majority of the changes to the topology require only slight modifications to the shortest path tree. This provides a strong argument for the use of an incremental shortest path calculation [1].

- An examination of specific anomalies that highlight impact of external routes on OSPF and the need for advanced network management tools. These anomalies lasted for significant periods of time without being resolved. They demonstrate the significant negative impact that customer routes can have on OSPF. They also demonstrate the need for advanced network management tools that are able to collect information from routing protocols and alert operators of these problems.

6.4 Mitigating Network Unavailability using Topology Aware Overlay Networks

Multi-homing and overlay networks are two widely studied approaches, aimed at leveraging the inherent redundancy of Internet's underlying routing infrastructure to enhance end-to-end application performance and availability. However, the effectiveness of these approaches depends on the natural diversity between paths taken by the packets injected into different provider networks. We conducted a detailed measurement-based study of the impact of Internet's dynamic path diversity on multi-homed and overlay networks. We based our year-long analysis on traceroutes and routing table data collected from several vantage points, including looking glasses at major Internet Service Providers (ISPs), BGP RouteViews servers and more than fifty PlanetLab nodes geographically distributed across the Internet.

Our experimental analysis is the first comprehensive study that quantifies several significant limitations of these architectures. Although a multi-homed stub network can choose among different upstream ISPs one hop away, it has no further control on the rest of the path to a given destination. Furthermore, our study exposes the limitations of current overlay networks that are constructed without explicitly considering the underlying Internet routing topology. Our results conclusively demonstrates that logically disjoint virtual links between overlay nodes—even placed in different autonomous systems with distinct administrative control—are likely to share physical links and routers at the IP layer.

These results may have significant ramifications on how the inherent topological redundancy of the underlying Internet infrastructure may be leveraged to enhance availability, performance and security. In order to take full advantage of multi-homing features, we need to keep each path as distinct as possible from others. However, since the Internet spans multiple administrative domains, achieving this goal is nearly impossible. The lack of global information and centralized control puts restrictions on exploiting the redundancy of the underlying Internet infrastructure.

In response to these observations, we developed a novel framework for topology-aware overlay networks that enhances the availability and performance of end-to-end communication. In this framework, we explicitly design overlay networks to maximize path independence without degrading performance. Based on our analysis of real-world networks, we use topology-aware node placement heuristics to ensure path diversity. This allows us to avoid path failures which are not avoidable using current existing overlay-based approaches. In addition, we validate this framework based on real Internet failures.

In addition to the overlay network framework, we developed the concept of a fault-tolerant virtual private network (FVPN)—a framework for supporting seamless network fail-over by leveraging the inherent redundancy of the underlying Internet infrastructure. The proposed architecture includes an application-level module, which is integrated into gateways at virtual private network (VPN) end-points. This module enables fail-over to a redundant path without waiting for the underlying routing protocol converging to a new route. Based on this framework, we have developed two heuristic algorithms for establishing redundant backup paths between two endpoints while minimizing overlapping network links. The FVPN framework does not require any changes to the network infrastructure and is easily supported

in today's commodity Internet. Furthermore, the framework is independent of routing policy of intermediate service providers. The framework was validated using empirical data from a collection of real-world data from MichNet's backbone network. This dataset represents both network topology and real-world faults for examining fault tolerant networking.

Our work with overlay networks is summarized below and presented in detail in three conference papers [9, 11, 10].

6.5 Impact of Path Diversity on Multi-homed and Overlay Networks

In recent years, a number of researchers have studied Internet routing protocols in terms of end-to-end behavior, convergence and stability [23, 18, 16, 4]. These studies have found that although the Internet routing infrastructure is highly redundant, current underlying routing protocols do not fully utilize this redundancy to achieve higher performance and availability goals. When an underlying routing protocol is slow to react and recover from the failure of a link or router, path failures are not transparently masked and are visible to end hosts. For instance, several studies including [23] observed that more than 20% of path failures are not recovered within 10 minutes. Such link or router failures may be visible because of delayed BGP convergence and/or fundamental forwarding problems (e.g., forwarding loops). BGP's fault recovery mechanisms sometimes take many minutes before routes converge to a consistent form [18, 16]. Furthermore, because current underlying routing protocols are restricted in flexibility, they lack the ability to detour around congested bottlenecked links. For example, BGP cannot detect performance problems such as persistent congestion on links, which affect end-to-end performance. As long as a link is live, BGP's routing will keep forwarding packets over the congested path.

Multi-homed and *overlay networks* are two widely discussed approaches [27, 22, 29, 3, 5] that utilize Internet redundancy to offer better performance and availability. Both approaches aim to provide alternate paths by exploiting path redundancy between endhosts. Multi-homing refers to a single network having more than one connection to the Internet. A stub network with connections to multiple providers may exhibit better performance and reliability than one with a single connection. Consider a situation where the customer is connected to both ISP1 and ISP2. If the ISP1-Customer link experiences congestion or failure, the traffic can be routed to go through the other link, the ISP2-customer link. Several commercial systems including Radware [27], netVmg [22], and RouteScience [29] attempt to provide enhanced availability or performance by leveraging the concept of multi-homing. These solutions are deployed in front of a multi-homed site, and they attempt to manage the site's redundant connections to its upstream service providers.

Other research projects such as RON [3] and Detour [5] leverage the topological redundancy of the Internet by constructing overlay networks to deliver better reliability and/or performance. An overlay network instantiates a virtual network on top of a physical network by deploying a set of overlay nodes above the existing IP routing infrastructure. Overlay nodes cooperate with each other to route packets on behalf of any pair of communicating nodes, forming an overlay network. If the underlying topology has physical path redundancy, it is possible to find alternative paths between overlay nodes when congestion or failure makes a primary path unavailable [31].

At first glance, systems based on multi-homed and overlay networks seem to be effective. It is widely believed that multi-homing and overlay networks can provide significant availability gains. However, the effectiveness of these systems depends on the assumption that paths available to packets traversing different ISPs (or overlay nodes) would enjoy a high degree of diversity and failure of each path should be independent of failure of alternate paths.

In reality, Internet path failures are indeed correlated. There are many factors that contribute to the dependency of path failure. For example, the failure of paths that travel across the same administrative

domain can be related to each other. Geographical adjacency can also be a factor. A failure at a Network Access Point (NAP) can affect all paths going through the NAP. Most of all, paths that share the same physical links and/or routers are very likely to experience failure at the same time.

This report quantitatively analyzes the impact of path diversity on multi-homed and overlay networks from several perspectives. It also highlights several inherent limitations of multi-homing and overlay architectures in fully exploiting the potential redundancy of the Internet [33, 35, 36]. We base our analysis on traceroutes and routing table data collected from several vantage points in the Internet including: *looking glasses* at ten major Internet Service Providers (ISPs), RouteViews servers [16, 30, 28] collecting routing data from twenty ISPs, and more than fifty PlanetLab nodes [24] distributed broadly across the Internet. The topological distribution of these collection points ensures that a broad range of upstream ISPs are represented in our study. The primary contributions of this study are as follows:

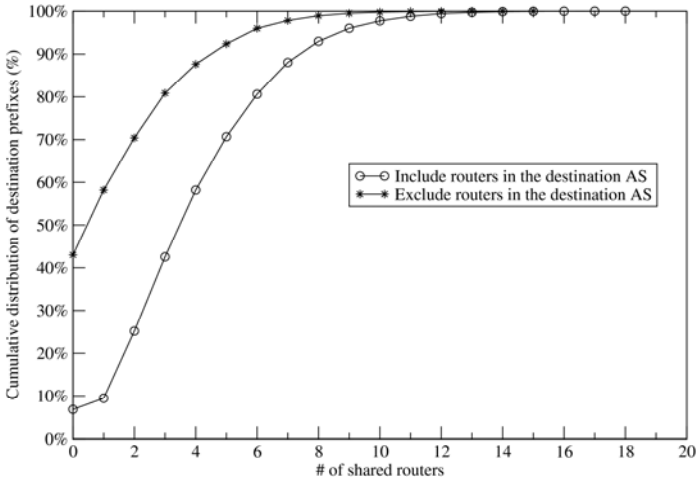


Figure 13: Link level overlapping from Ann Arbor.

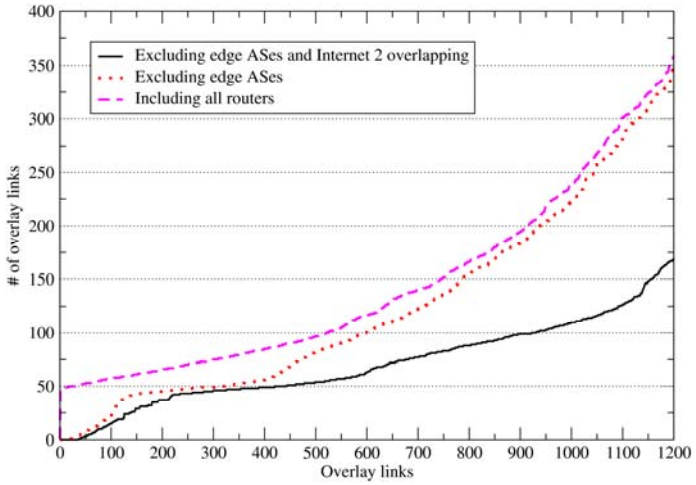


Figure 14: Overlapping among overlay links.

- We demonstrate significant limitations of multi-homing architecture from several perspectives:
  - We first quantify the extent of path diversity in multi-homed networks. These measurements reveal that a significant percentage of the paths from a multi-homed site may overlap. For example, when packets are injected from different ISPs to the same destination, the paths taken overlap at least once for 80% of 80,000 destinations networks in our study (Figure 13).

    - Our study also attempts to identify the source of overlapping for multi-homed networks. First, our results show that a significant percentage of paths from multi-homed sites merge in the core of the Internet. This result is consistent with prior research on Internet topology suggesting that the Internet core is formed by a mesh of tier-1 ISPs [33, 35]. Second, although a multi-homed stub network may choose its upstream providers, the stub network cannot necessarily detour around the shared infrastructures, because it has little control beyond the next hop. These findings motivate a necessity to incorporate topology considerations in designing multi-homed architectures.
    - Our results also show that carefully choosing a set of upstream ISPs cannot overcome inherent limitations of multi-homing. Although increasing the number of ISPs from 2 to 3 improves availability, having more than 3 upstream ISPs provides marginal gains in our study. Furthermore, even subscribing to as many as 10 upstream ISPs still results in at least one bottleneck router among multiple paths for 50% of destinations.
- Our analysis also exposes potential limitations of current overlay architectures:
    - We quantify the extent of correlation between overlay links. Our study reveals that logically disjoint virtual links between overlay nodes—placed in different Autonomous Systems (ASes) with distinct administrative control—are very likely to share links and routers at the IP layer (Figure 14).
    - Furthermore, our results show that most paths between overlay nodes and destinations also experience overlapping routers and links at the IP layer. In examining paths from overlay nodes placed inside various major ISPs to a set of over 80,000 destinations, we observe that even if overlay nodes are topologically diverse from each other, paths from different overlay nodes to the same destination typically share one or more intermediate ASes in addition to sharing the destination AS.
    - Our results contradict the conventional wisdom that placement of overlay nodes on different service provider networks would provide high degree of diversity. We observe that even if overlay nodes are deployed in various ISPs, overlay routes constructed without considering underlying topology may result in a significant degree of overlapping, and provide only limited availability gains.

6.6 Topology Aware Overlay Networks

In response our observations of the fundamental limitations of multi-homing and overlay architectures, we propose a novel framework for topology-aware overlay networks that enhances the availability and performance of end-to-end communication. In this framework, we explicitly design overlay networks to maximize path independence without degrading performance. To achieve this goal, we measure the diversity between different Internet Service Providers (ISPs) and also between different overlay nodes inside each ISP. Based on these measurements, we use topology-aware node placement heuristics to ensure path diversity. This allows us to avoid path failures which are not avoidable using current existing overlay-based approaches. The framework is based on our measurement-based analysis of traceroute and ping probes. These were collected from several vantage points in the Internet including *looking glasses* at ten major ISPs, and more than hundred PlanetLab nodes [24]. In addition, we validate this framework based on real Internet failures. The main contributions of this study are as follows:

- **A topology-aware overlay network framework to cope with path independence and improve availability and performance:** We explicitly design an overlay network to utilize path redundancy and maximize path independence between end hosts/networks. The proposed topology-aware overlay framework is a novel approach to increasing the availability and performance of end-to-end

communications. In the proposed framework, we deploy overlay nodes using off-line topology analysis rather than randomly deploying overlay nodes. Since operational topology change does not happen frequently,[1] this off-line node placement would only be updated over a long period as the Internet topology evolves. To accommodate transient topology changes due to congestion, link failures, or BGP instability, we provide flexibility in choosing overlay nodes on the fly, allowing our framework to successfully detour faulty or congested paths.

- **Topology-aware node placement heuristics:** We propose several strategies to deploy overlay nodes while considering the underlying topology. With the proposed measurement-based guidelines, we can identify which and how many ISPs we need to deploy overlay nodes at. For instance, we observe that choosing three out of ten ISPs provides a similar degree of path diversity and latency benefit as deployment of all 10 ISPs in our experimental setup. In addition, we also present *clustering-based* heuristics to select a subset of overlay nodes inside the same ISP to maximize topological diversity between the nodes. Our evaluation shows that this node placement approach is able to recover from significantly more path outages than existing overlay networks.
- **A simple, but effective routing mechanism on top of our topology-aware overlay architecture:** Our analysis results show that single-hop overlay paths provide the same degree of path diversity as multi-hop overlay paths for more than 90% of source and destination pairs. In addition, single-hop overlay paths improve latency for 90% of source/destination pairs compared with direct Internet paths. Therefore, we conclude that on top of topology-aware node deployment, single-hop overlay routing performs as well as multi-hop routing in terms of both availability and performance. In contrast to existing overlay networks [3], this single-hop overlay routing mechanism does not require a complicated routing protocol whereas existing overlay solutions impose large overhead, and therefore are less scalable. We also believe that by adding topology-aware node deployment, the single-hop routing mechanism becomes more effective in recovering from path outages and congestion problems.
- **Evaluation of our proposed approach using real-world data:** We validate this proposed framework based on real Internet failures. In this evaluation, we show that the proposed approach is able to react and recover from about 87% of path outages while the existing overlay networks only recovered from about 50% of path outages. We construct our evaluation platform using 232 points from 10 ISPs and 100 PlanetLab nodes. The topological distribution of these collection points ensures that a broad range of ISPs are represented in our study. The evaluation platform captures real-world failure events and also logs if the overlay paths could avoid this failure.

Overall, our study provides guidance to administrators and researchers on how to incorporate topology considerations in designing overlay architectures.

6.7 Fault-Tolerant Virtual Private Networks with An Autonomous System

Virtual Private Networks (VPNs) are a mechanism for providing secure and available communications at a commodity price. A Virtual Private Network (VPN) carves a “private” network over the public Internet, enabling a secure connection between hosts in multiple locations, creating a de facto private wide-area network for an enterprise. By overlaying private network connectivity on commodity Internet transit, organizations can significantly reduce their capital expenditures. This is in contrast to traditional private networks that require the purchase or leasing of dedicated communications infrastructure.

[1]Peering relationship between ISPs are changing with a very long period—months or even longer.

However, the trade off for cost is offset by the inherently fault prone Internet infrastructure. As the VPN becomes utilized for mission-critical processes, even short-lived failures can generate significant losses.

To address this issue, we present a framework for a *Fault-Tolerant VPN* (FVPN). The main idea is to pre-calculate and install a set of redundant minimally overlapping paths between the endpoints that can be used as backup routes as primary paths fail. This fail-over process of the FVPN is performed seamlessly, so that users and applications do not need to re-connect and re-authenticate. Within this framework, the FVPN can provide continuous VPN availability for multiple users and applications.

The main contributions of this work are the following:

- **Development of the Fault-Tolerant Virtual Private Network framework:** The FVPN framework is a novel approach to network availability. The approach trades off precomputation for fail-over latency. This trade off between connectivity and precomputation approaches the ideal of a real-time network fault oracle as the number of available disjoint routes increases. The experimental section demonstrates this result with as few as three backup routes.
- **Design of algorithms for generating sets of minimally overlapping backup routes:** Two heuristic algorithms are presented for constructing sets of maximally disjoint routes between two endpoints from a set of nodes and vertices.
- **Capture and replay of real-world link state failures in an operational service provider:** The framework is validated using empirical data from a collection of real-world data from an operational service provider backbone. This dataset represents both network topology and real-world faults for examining fault tolerant networking.

Our FVPN framework does not require any changes to the network infrastructure and is easily supported in today's commodity Internet. Furthermore, the framework is independent of routing policy of intermediate service providers.

7 Asymmetric Flows

We launched a new thread of research in October 2003 to analyze and quantify asymmetric flows on the Internet. This work identified the amount and types of asymmetry of Internet paths at both the AS and router levels. In addition, we defined the scope and impact of asymmetric traffic flows on Internet performance and security.

Data for this analysis was collected using two complementary techniques. First, we used traceroute probes from 56 Planetlab sites including mostly North American locations as well as a few European and Asian sites. We ran traceroute between each pair of hosts (approximately 1500 path pairs) once an hour. While traceroute provides detailed, router-level information about Internet routing paths it suffers from several limitations. First, packets can be lost due to congestion or filtering producing incomplete traces. In addition, traceroute measurement returns information about the router interfaces that saw a given packet, rather than information about the routers themselves. Determining if two interfaces belong to the same router is a difficult, and often inconclusive task.

To complement the traceroute measurements we also used data from several BGP collectors. These collectors provide live updates from several global BGP peering sessions. We used three sources of BGP data: University of Oregon's routeviews.org peers with 27 mostly North American routers; RIPE information from 13 routers mostly in Europe and Asia; and Arbor Networks and Merit Networks data providing mostly North American information. In contrast to traceroute data, BGP data provides AS-level information rather than router-level detail. However, BGP data also suffers from ambiguity. Policy decisions can affect the published AS path and also cause the actual path taken by packets to not reflect this published path. By combining data from both of these measurement techniques and presenting generalized results, we hope to overcome the low-level limitations.

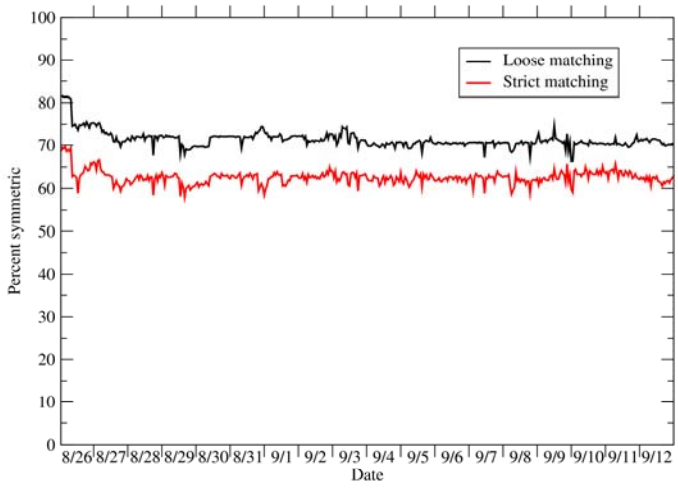


Figure 15: AS level asymmetry based on traceroute data.

The major results from this study are:

- AS-level asymmetry based on traceroute data shows that 65% to 75% of path-pairs are symmetric, with the amount of asymmetry relatively stable over time (Figure 15).
- This same data set shows that a third of the path-pairs are symmetric all the time, a sixth are asymmetric all the time, and most of the remaining path-pairs are asymmetric a significant fraction of the time.
- AS-level asymmetry based on BGP data shows that 60% to 70% of path-pairs are symmetric. In general, the European data shows more asymmetry than the North American data.
- Router-level asymmetry based on traceroute data shows about 5% to 80% of pairs are symmetric. This large range in the data is due to the ambiguities inherent in traceroute measurement. The lower bound includes only unambiguous results, while the upper bound all pairs that are not definitively asymmetric.

8 Tech Transfer

We are witnessing the proliferation of increasingly complex, widely distributed cyber attacks on commercial and mission-critical military systems interconnected by IP-based networks. Arguably chief among these security threats are Distributed Denial of Service (DDoS) attacks, zero-day network worms, and routing exploits. As illustrated by the DDoS attack on the Internet's thirteen root DNS servers, attackers have upped the ante by targeting the Internet's core infrastructure. No longer bound for individual web servers, the latest distributed attacks take a more insidious approach: they aim to disrupt the underlying Internet infrastructure itself. In addition, the increasingly global scale of threats coupled with the rapid evolution of attack technologies has rendered signature-based approaches to network security ineffective. Research from the Lighthouse Project led to the discovery of highly scalable, service provider-class solutions for the rapid detection, backtracking, and mitigation of DDoS attacks.

Working from a granular understanding of normal network flows, the anomaly detection system developed by this research can rapidly spot attacks, closing the costly gap between the detection of a

network-based security threat and its resolution. This novel, distributed anomaly-based detection scheme provides a pragmatic, highly- scalable early warning system for threats to mission-critical networks. It achieves this by regularly sampling network traffic statistics to establish a dynamic baseline of typical traffic patterns in different zones on the network. By comparing real-time network activity against this dynamic baseline, it is able to flag anomalies, including zero-day threats. Using precise fingerprints drawn from the detection and trace back processes, this technology recommends attack- and equipment-specific filters to eliminate the threat, ensuring normal operations. A highly successful deployment and demonstration of this technology on a production service provider network in the summer of 2000 led to the fast-paced transfer of this technology over that last four years.

This research has had a profound impact on the network security landscape during the last four years. The early success in demonstration and evaluation of the research concepts led to the rapid commercial availability of this technology by Arbor Networks, a network security company launched in 2001 with significant funding support from the private sector. By the end of 2003, security solutions based on this research had been deployed across mission-critical networks at the Department of Defense, more than 20 national and regional network service providers in North America, and numerous enterprises in the commercial sector. It is important to note that a significant portion of the Internet backbone is now protected by Arbor Networks technology (the Peakflow product line) that was first demonstrated in this DARPA/AFRL funded effort. Following a trajectory from its origins in DoD-sponsored university research to its current leadership as a provider of innovative solutions to government and industry, this partnership shines as an example of successful commercial availability of government-funded cyber security research.

The technology upon which the University of Michigan's Lighthouse project was built benefited from DARPA and AFRL support in several key ways. First and foremost the support was critical in providing the seed funding that led to this research breakthrough. This also enabled the research group to incubate a set of expertise capable of developing these new technologies including expertise in networking security, network engineering, and routing. In addition, access to Merit Networks provided a production service provider network as a platform for technology development: crucial given the nature of the problems the project set out to solve. This heritage has proved incredibly valuable as it influenced architecture, design, and features in ways that would be impossible to replicate in the lab. Indeed, much of the project's success over its competition can be traced to decisions made because of its production, service provider network heritage. This collaboration would not have been possible without the DARPA/AFRL sponsorship of a deployment and demonstration system on a production network.

The key features of Arbor Networks' Peakflow product include:

- Detection and Fingerprinting. Peakflow provides detection and detailed characterization (or fingerprints) to help network operators proactively flag known and new network threats.
- Traceback. Through network-wide comparison of recent anomaly fingerprints, Peakflow reconstructs the trajectory across the network, quickly identifying affected customers and equipment.
- Remediation. Using precise fingerprints drawn from the detection and trace back processes, Peakflow recommends attack and equipment specific filters to eliminate the threat, ensuring normal operations.
- Flexible reporting. Peakflow exports Extensible Markup Language (XML) anomaly data enabling network operators to easily build custom analysis for forensics, trending, and research.

Arbor is the Leader in Network Anomaly detection with unmatched market penetration. This has been enabled by several key-differentiating characteristics of the Peakflow technology:

- Scalability. Peakflow scales up to OC-192 speeds for network wide deployment.

- Anomaly-based. Peakflow's distributed architecture provides for real- time detection and resolution of zero day threats.
- Interoperable. Peakflow leverages the existing investments made in network devices.
- Non-intrusive. By supporting passive monitoring, Peakflow does not degrade the performance of the network.

This technology is deployed in over 50 major large networks across the globe protecting their operations from DDoS attacks, zero-day threats, routing attacks and traffic mismanagement. Beyond improved resiliency and security, Peakflow automates the painstaking detection, trace back, and remediation process, significantly lowering operational expense for large mission-critical networks.

In addition, the work summarized in this report has been published in papers written for public networking and security conferences targeting both researchers and network operators. In addition, we have debriefed a number of government practitioners and network operators on the results of our work.

8.1 Patents

Four patent disclosures were filed as part of this project based on the denial-of-service detection, back tracing, and filtering technologies from the Lighthouse project:

- Method and system for protecting publicly accessible network computer services from undesirable network traffic in real-time

  Application Number: 20020035698

  A method and system are provided for protecting publicly accessible network computer services from undesirable network traffic in real-time. The method includes receiving network traffic destined for the services and analyzing the network traffic to identify an undesirable user of the services. Access of the undesirable user to the services is limited to protect the services. The method and system identify and remove a new level of security threat that is not addressable by current techniques. Specifically, the method and system identify topologically anomalous application-level patterns of traffic and remove these data flows in real-time from the network.
- Method and system for detecting, tracking and blocking Denial of Service attacks over a computer network

  Application Number: 20020032871

  A system and method is provided for detecting, tracking and blocking Denial of Service (DoS) attacks, which can occur between local computer systems and/or between remote computer systems, network links, and/or routing systems over a computer network. The system includes a collector adapted to receive a plurality of data statistics from the computer network and to process the plurality of data statistics to detect one or more data packet flow anomalies. The collector is further adapted to generate a plurality of signals representing the one or more data packet flow anomalies. The system further includes a controller that is coupled to the collector and is adapted to receive the plurality of signals from the collector. The controller is constructed and arranged to respond to the plurality of signals by tracking attributes related to the one or more data packet flow anomalies to at least one source, and to block the one or more data packet flow anomalies using a filtering mechanism executed in close proximity to the at least one source.
- Method and system for reconstructing a path taken by undesirable network traffic through a computer network from a source of the traffic

  Application Number: 20020032793

A method and system for reconstructing a path taken by undesirable network traffic through a computer network from a source of the traffic is provided. The method includes collecting statistics at a plurality of measurement points located within forwarding infrastructure of the computer network. The method also includes analyzing the statistics to reconstruct the path taken by the undesirable network traffic through the network from the source of the traffic. The method and system use a combination of well-known misuse signatures of network resources in combination with modeling of normal network service behavior to identify bandwidth anomalies.

- Method and system for profiling network flows at a measurement point within a computer network

  Application Number: 20020032717

  A method and system for profiling network flows at a measurement point within a computer network is provided. The method includes measuring network flows having invariant features at a measurement point located within routing infrastructure of the computer network to obtain flow statistics. The method also includes aggregating the flow statistics to obtain a traffic profile of the network flows at the measurement point. The method and system utilize the natural hierarchy in the Internet addressing scheme to provide a means for making tractable measurements of network traffic in high-speed networks. Moreover, the method and system adapt dynamically to the changing underlying traffic characteristics to maintain a maximum memory footprint for the profiles. The method and system adapt by adjusting the level of aggregation of the traffic endpoints along a scale from Interface to fully specified network address.

References

[1] Cengiz Alaettinoglu and Steve Casner. A detailed analysis of ISIS routing protocol behavior. NANOG Presentation, February 2002.

[2] Guy Almes. Metrics and Infrastructure for IP Performance. http://io.advanced.org/csg-ippm/, September 1997.

[3] D. Anderson, H. Balakrishnan, F. Kaashoek, and R. Morris. Resilient overlay networks. In *Proceedings of 18th ACM Symposium on Operating Systems Principles*, 2001.

[4] A. Basu and J. G. Riecke. Stability issues in OSPF. In *Proceedings of ACM SIGCOMM*, 2001.

[5] A. Collins. The Detour framework for packet rerouting. Master's thesis, University of Washington, 1998.

[6] FreeBSD Homepage. `http://freebsd.org`.

[7] Fyodor. Remote OS detection via TCP/IP stack fingerprinting. `http://www.insecure.org/nmap/nmap-fingerprinting-article.html`, October 1998.

[8] Ramesh Govindan and Anoop Reddy. An analysis of inter-domain topology and route stability. In *Proceedings of the IEEE INFOCOM '97*, Kobe, Japan, April 1997.

[9] Junghee Han and Farnam Jahanian. Impact of path diversity on multi-homed and overlay networks. In *Proceedings of the International Conference on Dependable Systems and Networks (DSN)*, Palazzo dei Congressi, Florence, Italy, June 2004.

[10] Junghee Han, G. Robert Malan, and Farnam Jahanian. Fault-tolerant virtual private networks within an autonomous system. In *Proceedings of the IEEE Symposium on Reliable Distributed Systems (SRDS)*, Suita, Japan, October 2002.

[11] Junghee Han, David Watson, and Farnam Jahanian. Topology aware overlay networks. In *Proceedings of IEEE INFOCOMM*, Miami, FL, March 2005.

[12] Gianluca Iannaccone, C. Chuah, Richard Mortier, Supratik Bhattacharyya, and Christophe Diot. Analysis of link failures in an IP backbone. In *ACM SIGCOMM Internet Measurement Workshop*, Marseille, France, November 2002.

[13] Internet Performance Measurement and Analysis (IPMA) project. http://www.merit.edu/ipma/.

[14] IP Performance Metrics (IPPM). http://www.ietf.org/html.charters/ippm-charter.html.

[15] Intel Internet Exchange Architecture. `http://www.intel.com/design/network/ixa.htm`.

[16] Craig Labovitz, Abha Ahuja, Abhijit Bose, and Farnam Jahanian. Delayed Internet routing convergence. In *Proceedings of ACM SIGCOMM*, 2000.

[17] Craig Labovitz, Abha Ahuja, and Farnam Jahanian. Experimental study of Internet stability and wide-area network failures. In *Proceedings of FTCS99*, June 1999.

[18] Craig Labovitz, G. Robert Malan, and Farnam Jahanian. Internet routing instability. *IEEEACM Transactions on Networking*, 6(5):515–528, October 1998.

[19] Craig Labovitz, Roger Wattenhofer, Srinivasan Venkatachary, and Abha Ahuja. The impact of Internet policy and topology on delayed routing convergence. Technical Report MSR-TR-2000-74, Microsoft Research, 2000.

[20] Jamshid Mahdavi, Matt Mathis, and Vern Paxson. Creating a National Measurement Infrastructure (NIMI). Internet Statistics and Metrics Analysis (ISMA) ’97, May 1997.

[21] G. Robert Malan, David Watson, Farnam Jahanian, and Paul Howell. Transport and Application Protocol Scrubbing. In *Proceedings of the IEEE INFOCOMM 2000 Conference*, Tel Aviv, Israel, March 2000.

[22] netvmg. http://www.netvmg.com.

[23] V. Paxson. End-to-end routing behavior in the Internet. In *Proceedings of ACM SIGCOMM*, 1996.

[24] PlanetLab. http://www.planet-lab.org.

[25] Thomas H. Ptacek and Timothy N. Newsham. Insertion, Evasion, and Denial of Service: Eluding Network Intrusion Detection. Originally Secure Networks, Inc., now available as a white paper at the Network Associates Inc. homepage at http://www.nai.com/, January 1998.

[26] David Putzolu, Sanjay Bakshi, Satyendra Yadav, and Raj Yavatkar. The Phoenix Framework: A Practical Architecture for Programmable Networks. *IEEE Communications*, 38(3):160–165, March 2000.

[27] Radware. Linkproof: A traffic manager for multi-homed networks. http://www.radware.com.

[28] RIPE-NCC Routing Information. http://abcoude.ripe.net.

[29] RouteScience. http://www.routescience.com.

[30] RouteViews. http://www.routeviews.org.

[31] S. Savage, A. Collins, E. Hoffman, J Snell, and T Anderson. The end-to-end effects of Internet path selection. In *Proceedings of ACM SIGCOMM*, 1999.

[32] Aman Shaikh, Chris Isett, Albert Greenberg, Matthew Roughan, and Joel Gottlieb. A case study in OSPF behavior in a large enterprise network. In *ACM SIGCOMM Internet Measurement Workshop*, Marseille, France, November 2002.

[33] skitter project. http://www.caida.org/tools/measurement/skitter/.

[34] Matthew Smart, G. Robert Malan, and Farnam Jahanian. Defeating TCP/IP Stack Fingerprinting. In *Proceedings of 9th USENIX Security Symposium*, Denver, Colorado, August 2000.

[35] L. Subramanian, S. Agrawal, J. Rexford, and R. H. Katz. Characterizing the Internet hierarchiy from multiple vantage points. In *Proceedings of IEEE INFOCOM*, 2002.

[36] R. Teixeira, K. Marzullo, S. Savage, and G. M. Voelker. In search for path diversity in ISP networks. In *Internet Measurement Conference*, 2003.

[37] David Watson, Craig Labovitz, and Farnam Jahanian. Experiences with monitoring OSPF on a regional service provider network. In *Proceedings of the ICDCS 2003 Conference*, Providence, RI, May 2003. IEEE Computer Society.